SANDIA REPORT

SAND2007-5870
Unclassified Unlimited Release
Printed September 12, 2007

Use of Aria to Simulate Laser Weld Pool Dynamics for Neutron Generator Production

Patrick K. Notz, David R. Noble, Mario J. Martinez, Andrew M. Kraynik

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550

Sandia is a multiprogram laboratory operated by Sandia Corporation,
a Lockheed Martin Company, for the United States Department of Energy's
National Nuclear Security Administration under Contract DE-AC04-94-AL85000.

Sandia National Laboratories

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
U.S. Department of Energy
Office of Scientific and Technical Information
P.O. Box 62
Oak Ridge, TN 37831

Telephone: (865) 576-8401
Facsimile: (865) 576-5728
E-Mail: reports@adonis.osti.gov
Online ordering: http://www.osti.gov/bridge

Available to the public from
U.S. Department of Commerce
National Technical Information Service
5285 Port Royal Rd
Springfield, VA 22161

Telephone: (800) 553-6847
Facsimile: (703) 605-6900
E-Mail: orders@ntis.fedworld.gov
Online ordering: http://www.ntis.gov/help/ordermethods.asp?loc=7-4-0#online

SAND2007-5870
Unclassified Unlimited Release
Printed September 12, 2007

Use of Aria to Simulate Laser Weld Pool Dynamics for Neutron Generator Production

Patrick K. Notz, David R. Noble, Mario J. Martinez, Andrew M. Kraynik
Sandia National Laboratories
P.O. Box 5800
Albuquerque, NM 87185

Abstract

This report documents the results for the FY07 ASC Integrated Codes Level 2 Milestone number 2354. The description for this milestone is, "Demonstrate level set free surface tracking capabilities in ARIA to simulate the dynamics of the formation and time evolution of a weld pool in laser welding applications for neutron generator production." The specialized boundary conditions and material properties for the laser welding application were implemented and verified by comparison with existing, two-dimensional applications. Analyses of stationary spot welds and traveling line welds were performed and the accuracy of the three-dimensional (3D) level set algorithm is assessed by comparison with 3D moving mesh calculations.

Contents

List of Figures

Chapter 1

Introduction

1.1 Laser Welding

Laser welding is a fusion welding technique which is achieved by focusing a very high power density beam on a very fine spot. In contrast to other common welding techniques, laser welding does not depend solely on conduction for achieving weld penetration. Initially, a large percentage of the beam energy is reflected, since most metals are good reflectors. However, the absorbed energy quickly produces an energy absorbing ionized metal vapor, which rapidly accelerates the absorption of energy. The laser energy is not only absorbed on the surface, but to a depth into the workpiece. With high energy lasers, this leads to the development of a so-called keyhole – essentially a deep and narrow cavity drilled by the laser energy. Laser light is thought to be further scattered in the keyhole, thereby increasing the coupling of the laser energy into the workpiece.

Major attributes of laser welding are that it produces deep and narrow welds, with minimal collateral heating of workpieces, and it can be performed at high production rates. The laser can be narrowly and precisely focused, so that very precise and compact welds can be made, making laser welding ideal for joining miniature and intricate parts. Welds can be made very close to heat-sensitive components, such as glass-to-metals seals or other electronic circuits, while minimizing thermal distortions. Laser welding permits welding speeds of several meters per minute, with a total heat input that is much lower than other conventional techniques such as arc welding. These attributes make laser welding an ideal joining method in manufacturing of components and subsystems at SNL. Laser welding has become a key joining process used extensively in manufacturing and assembling of critical components in the neutron generator, as well as in current and future life extension programs for all the weapons systems. Nearly every current SNL component uses laser welds. SNL applications include fireset housings, inertial switches, fuses, detonators, and other explosive components.

Laser welding is a challenging multiphysics problem requiring complex 3D models utilizing massively parallel algorithms to enable high fidelity solutions. Major challenges in laser welds for weapons manufacturing include porosity formation and weld

morphology, especially large variations in weld shape/volume in the presence of surface active shield gases. Modeling and simulation in combination with experiment can provide an effective (often the only) means to understand and control laser welding to meet design specifications. Having a predictive capability for virtual welding is critical for enabling agile manufacturing, a responsive infrastructure and supports the SNL vision of a science-based engineering transformation of system development.

The aforementioned attributes notwithstanding, there remains much room for improvement of the laser welding process. The weld quality is strongly affected by operational settings, such as weld speed, pulse frequency (in pulsed laser welding), and shield gas. Some issues include laser coupling problems, porosity formation, and thermal distortions. In addition to solving these operational issues, the long-range goal of research in laser welding is to develop the ability to model weld morphology and weld integrity from basic principles.A fundamental goal of modeling is to provide understanding of how the altered grain structure of laser welds is associated with the final mechanical properties of the welded systems. Experiences such as mechanical failures occurring at welded joints indicate that the mechanical properties of welded material are often significantly different from unwelded material. The grain structures of welded and unwelded material are, not surprisingly, radically different and the grain structure often differs significantly at different portions of a weld. Neither the variation in the grain structures, nor the variation in mechanical properties is well understood.

The SNL laser welding program is aimed at advancing the understanding of the laser welding process via coupled experimentation and modeling. The modeling portion of this project has focused on building models that enable the weld thermal history to be predicted, including weld pool fluid dynamics with melting/solidification, latent heat effects and vapor recoil pressure. These numerical models provide a virtual welding capability which enables fluid and thermal designers and analysts to better predict weld joint shape and solidification history, both necessary elements to predict microstructure.

1.2 Milestone

This report serves as the completion criteria for work performed on the Level 2 Milestone: "Modeling of Laser Welding for the Weapons Complex." The general objective was to establish the readiness of Aria algorithms for modeling the multiphysics issues present in laser welding. Advancement of laser welding capability requires modeling the 3D transient fluid dynamics of free surface physics in molten metal. Aria features that enable these simulations include: a) level set interface tracking b) adaptive mesh refinement, c) massively parallel processing, d) Newton iteration solution of nonlinear systems, and e) a flexible platform for advanced velocity/pressure coupling techniques.

Each of these features was demonstrated in this study and will be discussed in the remainder of the report.

Chapter 2

Governing Equations and Boundary Conditions

2.1 The Physical Model

In this report, we focus on the problem of a laser spot weld since it is representative of a broader class of joining processes. The problem is illustrated schematically in figure 2.1 where a laser is incident on a target material. In reality, as discussed in chapter 1, a large percentage of the beam energy is initially reflected. The absorbed energy, however, quickly produces an energy absorbing ionized metal vapor, which rapidly accelerates the absorption of energy. Computationally, we treat this complex process as a simple melting problem where the material transitions from solid to liquid state at a melting temperature.

A particular complexity of modeling the laser welding process is the phase change that occurs. It is well understood how to computationally model the response of a solid material to thermal fluctuations. Likewise, the physics that define fluid dynamics of a molten weld pool are also well understood. However, combining these two physical processes, especially the transition from one to the other, is not well understood. In particular, the state of stress of the liquid material is expressed as a function of the rate of deformation (rate of strain) whereas the solid state of stress is typically a function of the deformation (strain) as well as the rate of deformation. A constitutive model for the state of stress that accommodates phase change is, however, not known.

In this work, the situation is remedied by treating the entire medium as a liquid. In regions of the material where the temperature is below the melting point, the viscosity of the fluid is several orders of magnitude larger than that in the fluid so that the time scales of flow in the "solid" portion of the domain are much longer than the molten portions of the domain.

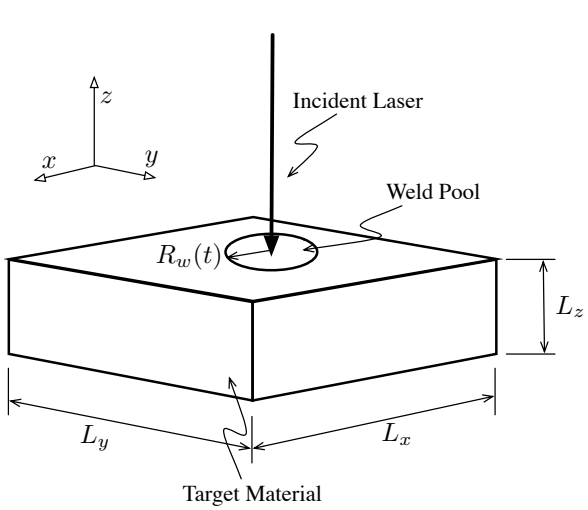

Figure 2.1. Schematic diagram of a laser spot weld. L_x, L_y and L_z are the geometry dimensions and R_w is the weld pool radius which is a function of time t.

2.2 Governing Equations

In this work, the fluid is treated as an incompressible liquid with velocity $\boldsymbol{v}$ and density ρ. The conservation of momentum is expressed with the Cauchy momentum equation,

$$\rho\frac{\partial \boldsymbol{v}}{\partial t} + \rho\boldsymbol{v}\cdot\boldsymbol{\nabla}\boldsymbol{v} = \boldsymbol{\nabla}\cdot\boldsymbol{\sigma} + \boldsymbol{f} \tag{2.1}$$

where t is time, $\boldsymbol{\sigma}$ is the fluid stress tensor and $\boldsymbol{f}$ is the body force. The constitutive model for the fluid stress is a generalized Newtonian fluid,

$$\boldsymbol{\sigma} = -p\boldsymbol{I} + \mu\left(\boldsymbol{\nabla}\boldsymbol{v} + \boldsymbol{\nabla}\boldsymbol{v}^t\right) \tag{2.2}$$

where p is the pressure and μ is the dynamic viscosity which may be a function of the temperature or shear rate. The only body force that needs to be considered in this work results from gravity so that $\boldsymbol{f} = \rho\boldsymbol{g}$ where $\boldsymbol{g}$ is the local gravitational acceleration.

The pressure is an additional unknown which is determined by satisfying the continuity equation which, for an incompressible liquid, reduces to

$$\boldsymbol{\nabla}\cdot\boldsymbol{v} = 0. \tag{2.3}$$

The temperature T throughout the material is determined by solving an energy transport equation which, for a single component liquid, is

$$\rho C_p\frac{\partial T}{\partial t} + \rho C_p\boldsymbol{v}\cdot\boldsymbol{\nabla}T = -\boldsymbol{\nabla}\cdot\boldsymbol{q} + H_v. \tag{2.4}$$

Here, C_p is the specific heat, $\boldsymbol{q}$ is the diffusive energy flux and H_v is the volumetric heat source. There are no volumetric sources of energy in this application so H_v is not considered in the remainder of this work. Fourier's law is used as the constitutive equation for the diffusive energy flux,

$$\boldsymbol{q} = -\kappa \boldsymbol{\nabla} T \tag{2.5}$$

where κ is the thermal conductivity.

2.3 Boundary Conditions

The physical domain is chosen to be large enough that zero velocity boundary conditions can be applied on the side and bottom surfaces (see figure 2.1). Furthermore, symmetry conditions are exploited whenever possible.

2.3.1 Free Space Radiation Energy Flux

Due to the large temperatures involved, the energy loss due to radiation must be accounted for. Here, we use a simple free-space radiation boundary condition,

$$q_r \equiv \boldsymbol{n} \cdot \boldsymbol{q}_r = \epsilon \sigma_b \left(T^4 - T_\infty^4 \right) \tag{2.6}$$

where σ_b is the Stefan-Boltzmann constant and T_∞ is the far-field temperature and ϵ is the absorptivity of the material.

2.3.2 Gaussian Spot Weld Energy Flux

The incident laser is taken to be a finite diameter circular beam with a Gaussian power distribution.

$$q_l \equiv \boldsymbol{n} \cdot \boldsymbol{q}_l = 2 r_{\text{eff}} F_\circ e^{-r_{\text{eff}} r_p^2 / R^2} \tag{2.7}$$

where R is the beam radius, r_{eff} is an effective radius, which can be used to specify the radial distribution of beam energy, and $F_\circ$ is the average heat flux of the laser.

2.3.3 Vapor Cooling Energy Flux

In order to account for the energy lost due to the ablation/evaporation of material during the laser welding process, a *vapor cooling* boundary condition is applied. The energy loss due the intense ablation of metal is modeled following the non-continuum molecular flow analysis of Knight (1979), with subsequent improvements by Kanouff

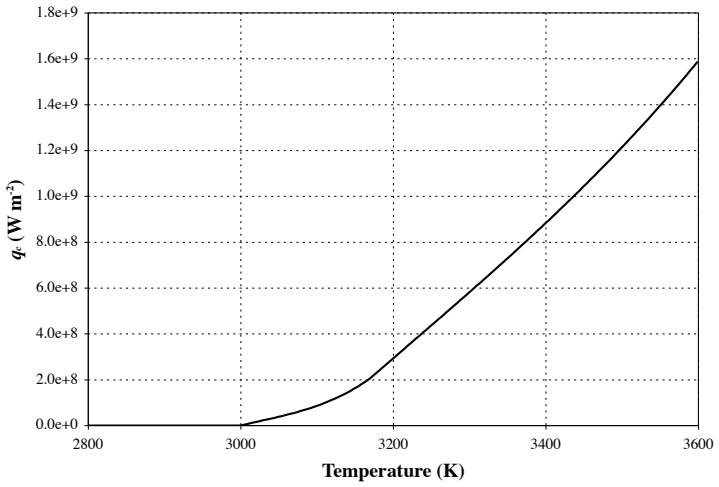

Figure 2.2. The vapor cooling energy flux $\boldsymbol{q}_c$ as a function of temperature.

(2003). The resulting energy flux is fit to a piece-wise cubic polynomial which is most conveniently expressed as a function of $\theta \equiv T - T_b$ where T_b is the boiling temperature. The function is split into three ranges of θ,

$$q_c = \boldsymbol{n} \cdot \boldsymbol{q}_c = \begin{cases} 0 & : \quad \theta < 0\,\mathrm{K} \\ a_{c,0} + a_{c,1}\theta + a_{c,2}\theta^2 + a_{c,3}\theta^3 & : \quad 0\,\mathrm{K} < \theta < 170\,\mathrm{K} \\ b_{c,0} + b_{c,1}\theta + b_{c,2}\theta^2 + b_{c,3}\theta^3 & : \quad \theta > 170\,\mathrm{K}. \end{cases} \tag{2.8}$$

In this work, the coefficients in 2.8 are $a_{c,0} = 0$, $a_{c,1} = 8.14373 \times 10^5$, $a_{c,2} = -2.24831 \times 10^5$, $a_{c,3} = 2.71683 \times 10^1$, $b_{c,0} = -3.1036 \times 10^8$, $b_{c,1} = 3.2724 \times 10^6$, $b_{c,2} = -1.8084 \times 10^3$ and $b_{c,3} = 2.7284$ and $\boldsymbol{q}_c$ has units of $\mathrm{W} \cdot \mathrm{m}^{-2}$. A plot of the vapor recoil pressure as a function of temperature is given in figure 2.2.

2.3.4 Vapor Recoil Pressure Momentum Flux

In order to account for the momentum transfer due to the evaporation of material during the welding process, a *vapor recoil pressure* boundary condition is applied. Like the vapor cooling boundary condition, this theoretical model was derived following Knight (1979) and Kanouff (2003); the recoil pressure is specified to be a polynomial in $\theta \equiv T - T_b$. The function is split into three ranges of θ,

$$p_r = \begin{cases} 0 & : \quad \theta < 0\,\mathrm{K} \\ a_{p_r,0} + a_{p_r,1}\theta + a_{p_r,2}\theta^2 + a_{p_r,3}\theta^3 & : \quad 0\,\mathrm{K} < \theta < 170\,\mathrm{K} \\ b_{p_r,0} + b_{p_r,1}\theta + b_{p_r,2}\theta^2 + b_{p_r,3}\theta^3 & : \quad \theta > 170\,\mathrm{K}. \end{cases} \tag{2.9}$$

In this work, the coefficients in 2.9 are $a_{p_r,0} = 0$, $a_{p_r,1} = 1.851502 \times 10^1$, $a_{p_r,2} = -1.969450 \times 10^{-1}$, $a_{p_r,3} = 1.594124 \times 10^{-3}$, $b_{p_r,0} = 0$, $b_{p_r,1} = -5.809553 \times 10^1$,

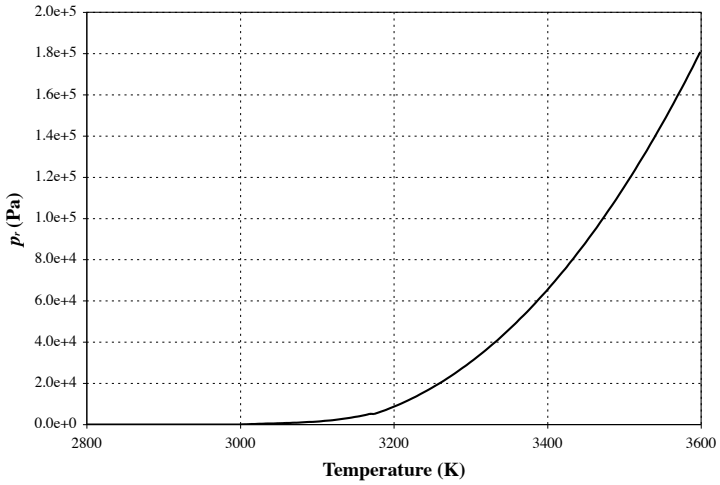

Figure 2.3. The vapor recoil pressure p_r as a function of temperature.

$b_{p_r,2} = 4.610515 \times 10^{-1}$, and $b_{p_r,3} = 2.332819 \times 10^{-4}$. The boiling temperature T_b is 3000K and p_r has units of Pa. A plot of the vapor recoil pressure as a function of temperature is given in figure 2.3.

2.3.5 Free Surface Boundary Conditions

The liquid/gas interface that exists at the surface of the weld pool is free to deform in time. The dynamics of this deformation are governed in part by the vapor recoil pressure but also by the surface tension of the interface. At such an interface, the jump in normal stress across the free surface due to the presence of surface tension σ is

$$\boldsymbol{n} \cdot (\boldsymbol{\sigma}_{liq} - \boldsymbol{\sigma}_{gas}) = -2\mathcal{H}\sigma\boldsymbol{n} - \boldsymbol{\nabla}_s\sigma. \tag{2.10}$$

Here, $-\mathcal{H}$ the local mean curvature of the surface and $\boldsymbol{\nabla}_s$ is the surface gradient operator, the subscripts *liq* and *gas* refer to the liquid and gas phases, respectively, and the normal vector $\boldsymbol{n}$ points from the liquid phase into the gas phase. The full numerical treatment of the boundary condition is well described in Cairncross et al. (2000) and will not be elaborated on here.

If one considers the gas to be a dynamically inactive and passive material then the stress vanishes in that phase except for a constant reference pressure which is typically taken to be zero. In that case, the stress balance in 2.10 reduces to

$$\boldsymbol{n} \cdot \boldsymbol{\sigma}_{liq} = -2\mathcal{H}\sigma\boldsymbol{n} - \boldsymbol{\nabla}_s\sigma. \tag{2.11}$$

Another observation can be made from equation 2.10. At equilibrium, the pressure is discontinuous at the free surface, viz.,

$$p_{gas} - p_{liq} = -2\mathcal{H}\sigma. \tag{2.12}$$

2.4 Material Models

2.4.1 Surface Tension

The surface tension is treated as a linear function of the temperature,

$$\sigma = \sigma_\circ - \alpha\,(T - T_\circ)\ \mathrm{N}\cdot\mathrm{m}^{-1} \tag{2.13}$$

where α is effectively $\boldsymbol{d}\sigma/\boldsymbol{d}T$ and $\sigma_\circ$ is the value of the surface tension at the reference temperature $T_\circ$.

2.4.2 Viscosity

The dynamic viscosity μ is a critical material property as was discussed in section 2.1. Here, we treat the viscosity as a piece-wise function of temperature using a fourth order polynomial for temperatures below the liquidus temperature, T_L, and a cubic polynomial for temperatures above T_L, viz.

$$\mu = \begin{cases} (c_{\mu,0} + c_{\mu,1}T_L + c_{\mu,2}T_L^2 + c_{\mu,3}T_L^3)\left[\beta + (1-\beta)\,\frac{T-T_{90}}{T_L-T_{90}}\right] & : \quad T < T_L \\ c_{\mu,0} + c_{\mu,1}\hat{T} + c_{\mu,2}\hat{T}^2 + c_{\mu,3}\hat{T}^3 & : \quad T \geq T_L. \end{cases} \tag{2.14}$$

were $\hat{T}$ is $\max(T, T_{\max})$ and μ has the units of $\mathrm{Pa}\cdot\mathrm{s}$. The nominal parameters used in this study are $T_{\max} = 4000\,\mathrm{K}$, $T_L = 1623\,\mathrm{K}$, $T_{90} = 1528\,\mathrm{K}$, $\beta = 10^{11}$, $c_{\mu,0} = 1.5616\times10^{-1}$, $c_{\mu,1} = -3.3696\times10^{-5}$, $c_{\mu,2} = 1.0191\times10^{-8}$, and $c_{\mu,3} = -1.0413\times10^{-12}$. This viscosity function is shown graphically in figure 2.4.

2.4.3 Thermal Conductivity

The thermal conductivity κ is taken to be a linear function of temperature,

$$\kappa = a_{\kappa,0} + a_{\kappa,1}T\ \mathrm{W}\cdot\mathrm{m}^{-1}\cdot\mathrm{K}^{-1} \tag{2.15}$$

with $a_{\kappa,0} = 10.7143$ and $a_{\kappa,1} = 14.2857\times10^{-3}$ and T is in Kelvin.

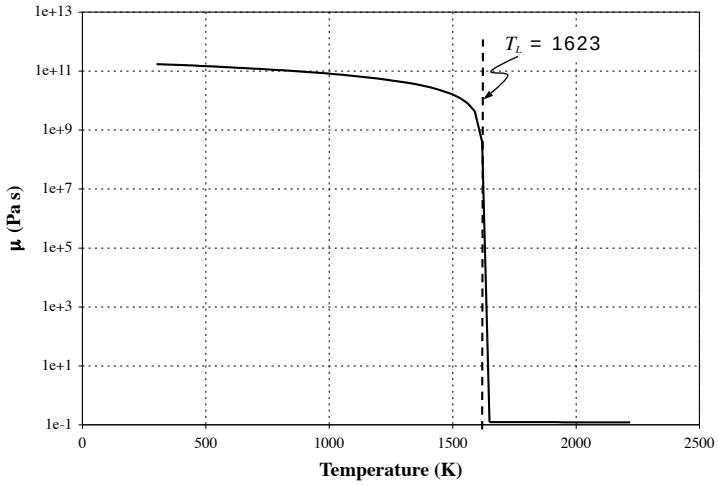

Figure 2.4. The viscosity μ as a function of temperature. The large values of μ at lower temperatures simulate the unmelted steel.

2.4.4 Specific Heat

The specific heat C_p is taken to be a linear function of temperature,

$$C_p = a_{C_p,0} + a_{C_p,1} T \text{ J} \cdot \text{kg}^{-1} \cdot \text{K}^{-1} \tag{2.16}$$

with $a_{C_p,0} = 425.75$ and $a_{C_p,1} = 170.833 \times 10^{-3}$ and T is in Kelvin.

2.4.5 Density

The density ρ is taken to be a constant, $\rho = 7.9 \times 10^3 \text{ kg} \cdot \text{m}^{-3}$.

Chapter 3

Computational Algorithms

3.1 Interface Tracking

A key requirement in the numerical analysis of this problem is tracking the location of the melt/air interface in time. Several methods exist for doing this numerically and in the continuum mechanics literature these fall into two classes: conformal discretization and interface tracking. Naturally, each of these approaches comes with its own benefits and detriments.

Examples of conformal discretization methods include Lagrangian and Arbitrary Lagrangian-Eulerian (ALE) methods. Here, the typical approach is to solve an auxiliary system of PDEs for the unknown coordinates of the mesh points (see, e.g., Cairncross et al. (2000)) so the mesh deforms with the material and points on the boundary of the mesh remain tied to the boundary of the material. The benefit of this approach is that boundary conditions can be applied in the usual way. Fields and equations always live at the same mesh points for all time so the algorithm is easy to implement. Lastly, one only needs to discretize the material of interest and not portions of space that the material may occupy at some time during the analysis. One problem with conformal techniques, however, is that they tend to be computationally expensive. For example, a three dimensional problem may add up to three additional degrees of freedom at each mesh point, sometimes nearly doubling the total number of degrees of freedom in the problem. A more serious problem, however, is that severe deformations lead to mesh entanglement and, hence, remeshing is necessary. This additional cost and complexity makes conformal interface tracking impractical for problems like laser welding.

Interface tracking techniques, such as level set methods (see, e.g., Sethian (1999)) and volume of fluid (see, e.g., Hirt and Nicholls (1981)) take a much different approach to the problem. In interface tracking methods the location of the moving interface is typically tied to an isovalue of a scalar field. Thus, tracking the material boundaries is more computationally efficient since one only needs to solve a scalar axillary equation. The mesh is fixed in time and the material moves freely through the mesh which makes problem setup very simple and the analysis doesn't require remeshing no matter how complex the interface deformations are. The lack of a dis-

crete boundary representation for the free surface, however, means that free surface boundary conditions must be handled differently. Moreover, since the field discretizations are non-conformal with the free boundary, it's not always possible to specify the boundary conditions accurately. Though it's not a requirement of interface tracking techniques, many implementations require all fields and equations to be defined over the entire mesh and that their values be computed at all times. This causes additional computational expense and, often times, computational difficulty due to scaling and resolution problems.

3.2 The Level Set Method

Due to the computational efficiency and the easy of use, our work here focuses on using the level set method for tracking the free surface. However, ALE simulations for problems with mild deformations are used as a means of assessing the accuracy of the level set simulations. In this section, we provide the details of the level set algorithm as it is implemented in Aria and used in this work.

First, we define a signed distance function $F(\boldsymbol{x}, t)$ that is a function of space $\boldsymbol{x}$ and time t and whose magnitude is the shortest distance from $\boldsymbol{x}$ to any point on the free surface, viz. the contour $F = 0$ defines the free boundary tracked by F. The sign of F is used to indicate whether the point $\boldsymbol{x}$ lies inside the material. Notationally, we denote these two sides of the interface as "phase A" and "phase B" and often use "A" and "B" as subscripts on mathematical quantities that are specific to a phase. Lastly, by convention, we associate phase A with $F < 0$.

In order to perform integrations over each phase, it's convenient to introduce a Heaviside function $H_i(F)$ which is unity in phase i and zero in the other phase. Moreover, H_A and H_B are related as $H_B = 1 - H_A$. In so-called diffuse interface approaches, the Heaviside function is regularized so that there is a smooth transition from one phase to another. In this work, we use

$$H_B(F) = \begin{cases} 0 & : \quad F < -\frac{\alpha}{2} \\ \frac{1}{2}\left[1 + \frac{F}{\alpha/2} + \frac{1}{\pi}\sin\left(\pi\frac{F}{\alpha/2}\right)\right] & : \quad -\frac{\alpha}{2} < F < \frac{\alpha}{2} \\ 1 & : \quad F > \frac{\alpha}{2} \end{cases} \tag{3.1}$$

where α is defined as the width of the interfacial region. Likewise, one can define a regularized delta function as,

$$\delta(F) = \frac{\mathrm{d}H(F)}{\mathrm{d}F} = \begin{cases} 0 & : \quad F < -\frac{\alpha}{2} \\ \frac{|\boldsymbol{\nabla} F|}{\alpha}\left[1 + \cos\left(\pi\frac{F}{\alpha/2}\right)\right] & : \\ & : \quad -\frac{\alpha}{2} < F < \frac{\alpha}{2} \\ 0 & : \quad F > \frac{\alpha}{2} \end{cases} \tag{3.2}$$

Figure 3.1 illustrates $H_B(F)$ and $\delta(F)$ for the case of $\alpha = 1$.

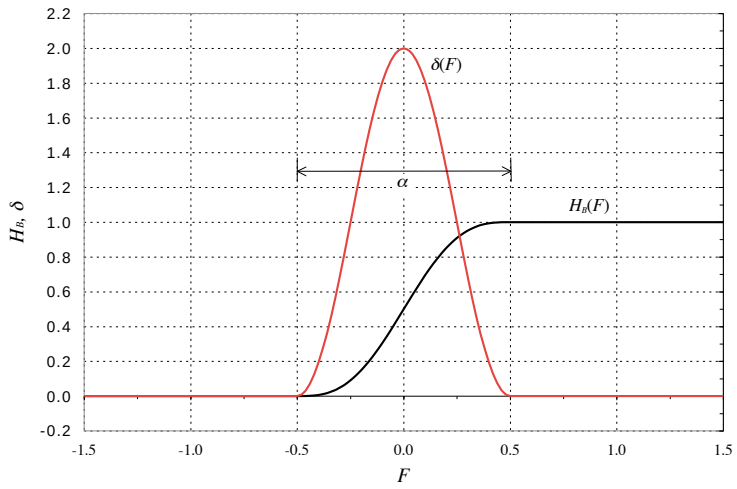

Figure 3.1. A plot of the regularized Heaviside (Eq. 3.1) and delta (Eq. 3.2) functions for an interfacial width $\alpha = 1$.

The distance function F must evolve in time as a part of the simulation in order to track the free surface location. Depending on the physics involved in a problem there may be different conditions which can be used to evolve the free surface in time. For example, in a phase change problem, the net normal flux of mass may define the interfacial velocity and hence its motion. In two phase flow problems, such as the laser welding application under study in this work, F is typically evolved using a transport equation,

$$\frac{\partial F}{\partial t} + \boldsymbol{v}_e \cdot \boldsymbol{\nabla} F = 0 \tag{3.3}$$

where $\boldsymbol{v}_e$ is the so-called *extension velocity*. The extension velocity is a vector field defined as the velocity which preserves the distance quality of F. Unfortunately, this field is not known and, in general, may be multi-valued or singular at some points in space and time. The typical approach is to instead use the fluid velocity $\boldsymbol{v}$,

$$\frac{\partial F}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} F = 0. \tag{3.4}$$

A consequence of this compromise is that distance quality of F will degrade in time and so *redistancing* or *renormalizing* algorithms must be run periodically to correct F. In this work, we utilize a constrained redistancing algorithm that attempts to redistance F so that the volume of each phase remains unchanged.

Note that equation 3.4 is purely hyperbolic. With the initialization of F and condition that the free surface defines the $F = 0$ isosurface, boundary conditions are not required.

3.3 ALE Mesh Motion

As described above, conformal moving-mesh algorithms have distinct advantages that make them ideal for verifying the accuracy of level set simulations. Furthermore, since Aria has both ALE and level set capabilities we will use ALE simulations as the basis for assessing the performance and accuracy of the more generally applicable level set simulations.

The ALE approach used in Aria is the pseudo-solid approach described by Cairncross et al. (2000). Here, the mesh is treated as though the nodes were material points of a solid that deforms quasi-statically. In Aria, we use a hyperelastic material description where the material coordinates $\boldsymbol{X}$ in the undeformed reference frame are related to the physical coordinates $\boldsymbol{x}$ in the physical reference frame by a displacement field $\boldsymbol{d}$, viz.,

$$\boldsymbol{x} = \boldsymbol{X} + \boldsymbol{d} \tag{3.5}$$

In this case, the Cauchy momentum equation that governs the quasi-static deformation of this pseudo-solid reduces to

$$\boldsymbol{\nabla} \cdot \boldsymbol{\sigma}_m = \boldsymbol{0} \tag{3.6}$$

where $\boldsymbol{\sigma}_m$ is the mesh (solid) stress tensor. Due to the large deformations that can occur we choose here to use a nonlinear neo-Hookean elastic constitutive equation. First, it's convenient to define the deformation gradient $\boldsymbol{F}$,

$$\begin{aligned} \boldsymbol{F} &\equiv \boldsymbol{\nabla}_X \boldsymbol{x}^t \\ &= \boldsymbol{\nabla}_X \boldsymbol{X}^t + \boldsymbol{\nabla}_X \boldsymbol{d}^t \\ &= \boldsymbol{I} + \boldsymbol{\nabla}_X \boldsymbol{d}^t \end{aligned}$$

where $\boldsymbol{\nabla}_X$ is the gradient operator in the undeformed reference frame. Then, the neo-Hookean elastic constitutive equation can be written as

$$\boldsymbol{\sigma}_m = \frac{\mu}{J}(\boldsymbol{b} - \boldsymbol{I}) + \frac{\lambda}{J} \ln J \boldsymbol{I} \tag{3.7}$$

where λ and μ and the Lamé coefficients, $\boldsymbol{b} \equiv \boldsymbol{F} \cdot \boldsymbol{F}^t$ is the left Cauchy-Green tensor and $J \equiv \det \boldsymbol{F}$. See, e.g., Bonet and Wood (1997) or Belytschko et al. (2004).

The boundary conditions on the mesh equation are very straight forward. On external boundaries, we prohibit normal displacements, $\boldsymbol{n} \cdot \boldsymbol{d} = 0$, but leave the mesh points free to move tangentially in order to minimize mesh distortions. Along the free surface, we use the kinematic boundary condition,

$$\boldsymbol{n} \cdot \left(\frac{\mathrm{d}\boldsymbol{d}}{\mathrm{d}t} - \boldsymbol{v} \right) = 0 \tag{3.8}$$

which states that material points on the surface remain on the surface.

3.4 The Finite Element Method

The system of equations to be solved include the conservation equations 2.1, 2.3 and 2.4 and the axillary equations for either interface tracking 3.4 or the mesh displacements 3.6. Respectively, the unknown fields in the problem are the velocity $\boldsymbol{v}$, pressure p, temperature T and either the distance function F or the mesh displacements $\boldsymbol{d}$. We solve this system of equations, subject to the boundary and initial conditions outlined above, using the Galerkin finite element method (G/FEM). G/FEM is a well established method for solving boundary value problems so we only describe it briefly here; full details of the method are available in Hughes (2000).

Each unknown field X is discretize by defining a set of basis functions $\{\phi_X\}$ and coefficients $\{X\}$ such that,

$$X(\boldsymbol{x}, t) = \sum_{i=1}^{N_X} X_i(t)\phi_X^i(\boldsymbol{x}) \tag{3.9}$$

where N_X is the number of basis functions and coefficients in the discretization. The basis functions $\{\phi_X\}$ are selected to be low order polynomials with local support.

Weak residuals of the governing PDEs are formed using the basis functions as the weighting functions. The residual form for the governing PDEs are summarized in equations 3.10–3.14.

$$R_{m,k}^i = \int_{\mathrm{V}} \left[\left(\rho \frac{\partial \boldsymbol{v}}{\partial t} + \rho \boldsymbol{v} \cdot \boldsymbol{\nabla} \boldsymbol{v} - \rho \boldsymbol{g} \right) \cdot \boldsymbol{e}_k \phi^i + \boldsymbol{\sigma}^t : \boldsymbol{\nabla} \left(\boldsymbol{e}_k \phi^i \right) \right] \mathrm{dV} - \int_{\mathrm{S}} \boldsymbol{n} \cdot \boldsymbol{\sigma} \cdot \boldsymbol{e}_k \phi^i \, \mathrm{dS} = 0 \tag{3.10}$$

$$R_P^i = - \int_{\mathrm{V}} \boldsymbol{\nabla} \cdot \boldsymbol{v} \phi^i \, \mathrm{dV} = 0 \tag{3.11}$$

$$R_T^i = \int_{\mathrm{V}} \left[\left(\rho C_p \frac{\partial T}{\partial t} + \rho C_p \boldsymbol{v} \cdot \boldsymbol{\nabla} T \right) \phi^i - \boldsymbol{\nabla} \phi^i \cdot \boldsymbol{q} \right] \mathrm{dV} + \int_{\mathrm{S}} \boldsymbol{n} \cdot \boldsymbol{q} \phi^i \, \mathrm{dS} = 0 \tag{3.12}$$

$$R_F^i = \int_{\mathrm{V}} \left(\frac{\partial F}{\partial t} + \boldsymbol{v} \cdot \boldsymbol{\nabla} F \right) \phi^i \, \mathrm{dV} = 0 \tag{3.13}$$

$$R_{m,k}^i = \int_{\mathrm{V}} \boldsymbol{\sigma}_m^t : \boldsymbol{\nabla} \left(\boldsymbol{e}_k \phi^i \right) \, \mathrm{dV} = 0 \tag{3.14}$$

Time derivatives are discretized using a first-order finite-difference (Euler) approximation. We use an explicit forward Euler approximation for the first four time steps followed by an implicit backward Euler approximation for subsequent time steps.

With this discretization in place, what remains is a large system of discrete nonlinear equations. This system is solved using Newton's method with analytic sensitivities for the Jacobian matrix.

3.5 Pressure Stabilization

For the finite element solution of the Navier-Stokes equations, there exist mathematical restrictions on the choice of basis functions for the velocity and pressure degrees of freedom. The details of this restriction are beyond the scope of this report but may be found elsewhere (see, e.g., Hughes 2000). In order to obtain solutions to the problem, one must either use a compatible pair of finite element basis functions or add stabilization terms to relieve the mathematical restrictions. For the sake of computational efficiency and geometric accuracy for the level set algorithms we prefer to use linear, nodal, C_0-continuous basis functions for all of our unknown fields. However, in order to do so we must employ some form of velocity-pressure stabilization.

Here, we employ the stabilization technique developed by Dohrmann and Bochev (2004). In addition to being computationally efficient and easy to implement, this stabilization method is also easy to use since it requires no special treatment for boundary conditions. Numerically, this method simply requires an additional volume integral term in the weak form of the continuity equation (equation 3.11).

3.6 Adaptive Mesh Refinement

A key to obtaining accurate simulation results with level set methods is sufficient resolution of the interfacial region. Since the interface typically propagates through the mesh in time, the mesh needs to have sufficiently small elements where ever the interface may pass. This worst-case approach to mesh refinement can lead to very computationally expensive simulations, especially in 3D. A remedy to this problem is to use *adaptive mesh refinement* (AMR) or h-adaptivity. The Aria application is built on top of the SIERRA framework (Edwards 2002) and so inherits an adaptive mesh refinement capability (Steward and Edwards 2002). The AMR capability allows some of the elements in a mesh to be refined (subdivided) and later unrefined to as to produce a locally refined mesh.

There are a two basic elements that contribute to the overall AMR algorithm in Aria. The first is that elements are *marked* for refinement according to some criterion that the user specifies. This step includes imposing a 2:1 refinement constraint so that an element's neighboring elements are never split more than once (see Steward and Edwards 2002). The second step is the actual refinement or unrefinement of elements. This step includes the prolongation of the unknown fields to new nodes in the mesh and the addition of hanging node constraints to the linear system to account for any nonconformal element boundaries. In order to use AMR an analyst only needs to specify a element marking function.

In this work, we use an element marking function that ensures that elements within a specified distance of the $F = 0$ interface have a size that is smaller than a specified

length scale. Since the interfacial width α is already supplied as part of the level set algorithm, it's convenient then to specify the desired number of elements across this width, n_α, so that the element length-scale in the interfacial region will be $h_I \equiv \alpha/n_\alpha$. In order to avoid excessive refinement and unrefinement at each time step, this marking criterion is applied to all elements where $|F| < \alpha/2 + \frac{3}{2}h_I$ at some point within the element.

It is important to note that the accuracy and efficiency of the level set algorithm is heavily dependent on the parameters α and n_α. If either α or n_α is too large then the diffusive nature of the level set algorithm can lead to significant losses of mass, momentum and energy. Also, if α/n_α is too large then the interfacial contributions to 3.10–3.14 will be under-resolved and, again, lead to inaccuracies. Lastly, if n_α is large the problem becomes very computationally expensive. Experience has lead to a recommended value of $n_\alpha = 4$. Recommended values of α, however, are very problem dependent. In this work, α is most closely correlated with the radius of the laser and is usually on the order of 10% of the beam radius.

Chapter 4

Results

4.1 Comparisons Between Goma and Aria Weld Simulations

Verification of the weld model implementation in Aria was accomplished by comparison with the Goma weld model results for a 2D spot weld. For the sake of simplicity, surface deformations were ignored for this simulation. Figure 4.1 shows the weld pool, velocity vectors, and temperature distribution after a 1 ms 200 W laser pulse, with a 0.1 mm spot radius. This power level produces a bowl-shaped melt pool with counter-rotating convection cells driving surface flows from the center of weld to the edges. The flow direction is determined by the temperature-dependent surface tension, which generally decreases with temperature for most metals. The contour marks the liquidus temperature.

Figures 4.2 and 4.3 compare the shear velocity on the weld pool surface and surface temperature between Goma and Aria simulations. The close comparison verifies the model implementation in Aria, especially given the complexity of the physics being modeled.

4.2 Comparisons with Spot Weld Experiments

Experiments on spot welds performed by J. Norris and C Robino (SNL, Joining and Coating Department, 1813) provide an opportunity for validation of the Aria weld model. The set of experiments considered here involve a 3.3 J/6 ms laser power setting, with a 350 micron beam radius on 304L stainless steel. The welds were performed using several different shield gases, including argon, nitrogen and air. The shield gas has a profound effect on the weld profile, depending of whether it provides an inert (argon) or oxidizing (air) atmosphere.

Figures 4.4 and 4.5 compare the weld profiles using argon and air as the shield gases, respectively. These are results from two-dimensional simulations whereas the welds are axisymmetric at this laser power. The experimental weld pool profiles are shown

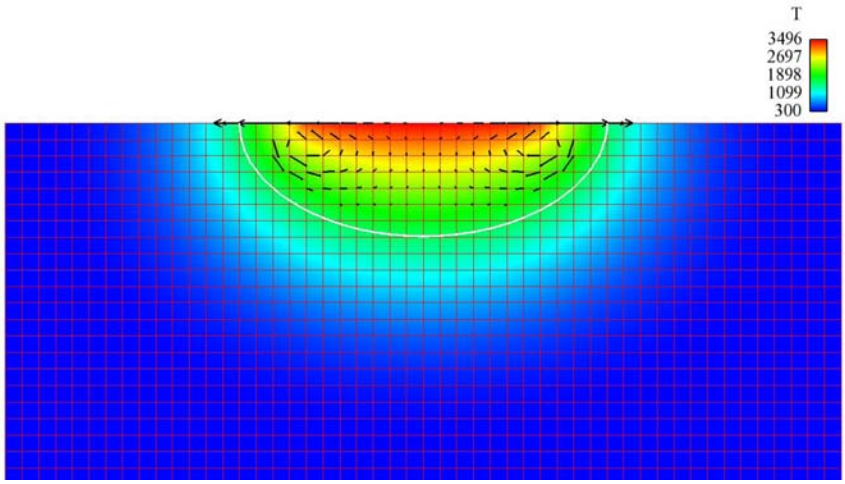

Figure 4.1. Weld pool shape and temperature distribution (2D) at 1 ms in stainless steel SS304L.

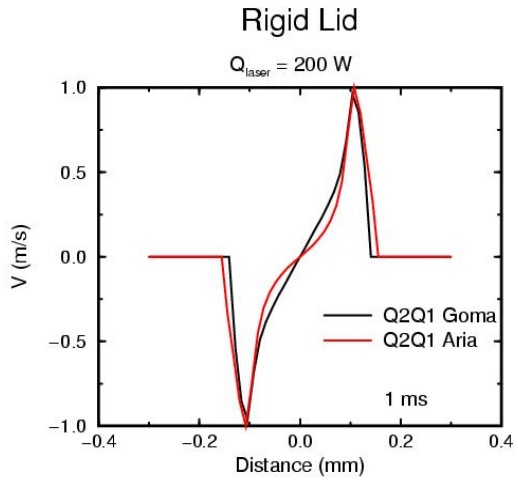

Figure 4.2. Comparison of surface shear velocity components at 1 ms for a 2D spot weld simulation.

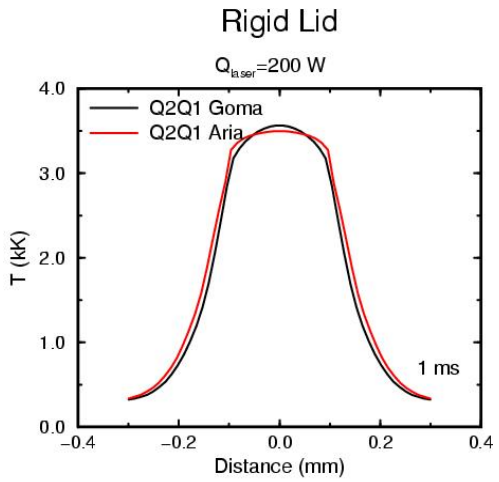

Figure 4.3. Comparison of surface temperature profile at 1 ms for a 2D spot weld simulation.

in red and the computed temperature contours in black. The color fringe depicts the temperature distribution as 6 ms. Also shown are instantaneous streamlines. The shield gas has a profound effect on the weld profiles. The surface tension of metals generally decreases with temperature. However, it is well known that dissolution of impurities, as might be introduced by oxidation, profoundly change the character of surface tension, generally significantly reducing the value near the melt point and increasing with temperature thereafter. In the simulations, with argon as the shield gas, the surface tension specified represents pure Fe. In this case the convective circulation on the surface is from the center towards the edges. This pattern brings hot molten metal to the periphery of the weld, which in the simulations promotes the lateral melting of the metal, resulting in a shallow weld pool. In the figure, the circulation exists only at the edges of the weld, while the central portion is largely quiescent. The absorptivity was set at 25%, resulting in a width/depth of 734/152 microns, compared with 675/270 in the experiments.

With air as shield gas the surface tension was modeled after the idealized gradient used in Tanaka and Lowke (2007) for SUS 304 with high sulfur concentration ($\sigma_\circ = 1.36$ N/m; $\alpha = -2.1 \times 10^{-4}$ N/m/K; see equation 2.13. Here the flow on the surface is from the edges of the weld pool towards the center, resulting in a downward directed jet of molten metal in the central portion of the weld pool. The positive surface tension gradient results in deeper welds because the circulation convects hot molten metal downward towards the bottom of the weld pool.

Using a fixed absorptivity of 30%, the figure shows reasonably good comparison with the experimental profile. The computations result in a width/depth of 640/261 (microns) compared to the experimental values of 675/270.

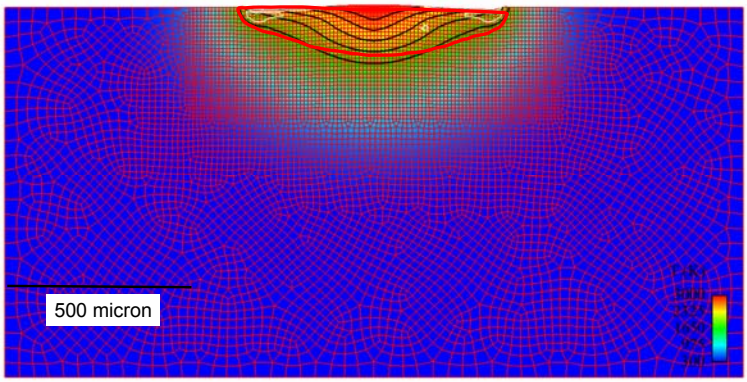

Figure 4.4. Weld profiles in stainless steel SS304L with Argon as shield gas. The profile in red is from the experiments which is compared to the temperature profiles in black. The coolest isotherm depicts liquidus.

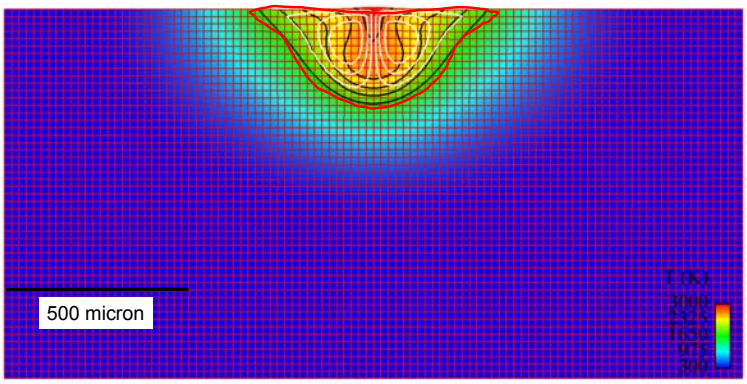

Figure 4.5. Weld profiles in stainless steel SS304L with air as shield gas. The profile in red is from the experiments which is compared to the temperature profiles in black. The coolest isotherm depicts liquidus.

4.3 Comparisons Between ALE and Level Set Algorithms

As noted in chapter 3 there are distinct advantages for both the ALE (conformal mesh) and level set (interface tracking) algorithms for simulating free surface flows. The level set algorithm is desirable because of its ease of use and its ability to capture large deformations and topological changes in deforming media. In this section we make comparisons between level set simulations and ALE simulations.

4.3.1 Effect of Including the Gas Phase

When the gas phase above the weld pool free surface is included in the simulation the pressure and velocity fields are treated as continuous between the two phases. Due to the presence of the gas, the stress balance at the free surface includes the stresses from the air phase. As discussed in section 2.3.5, the pressure is actually discontinuous at the free surface, in proportion to the surface tension and curvature of the free surface,

$$p_{gas} - p_{liq} = -2\mathcal{H}\sigma. \tag{4.1}$$

Equation 4.1 makes clear that in the presence of surface tension, the pressure is discontinuous at the interface. However, in the current level set algorithm all fields, including the pressure, are continuous across the interface. The result of this pressure continuity is that an artificial pressure gradient normal to the interface exists and leads to a mass flux across the interface. This effect can be seen by comparing three ALE simulations. In the first simulation, illustrated in figure 4.6, the gas phase is not simulated and so there is no artificial pressure gradient at the free surface and so mass is conserved; the volume of fluid displaced from the weld pool accumulates in a bulging ring around the edge of the weld.

In the second ALE simulation, illustrated in figure 4.7, we include the ambient gas phase which results in a mass loss in the liquid phase due to the artificial pressure gradient at the interface.

In the final ALE simulation, we include the gas phase in the simulation but use two different pressure fields, one for each phase, so that the pressure is discontinuous across the gas/liquid interface. In this case, figure 4.8, volume is again conserved since there is no artificial pressure gradient at the interface.

The level set algorithm currently available in Aria does not support discontinuous fields across the interface. Accordingly, this mass loss effect is observed for level set simulations. However, the amount of artificial mass transfer is proportional to the volume of the elements along the free surface (where the gradient exists). Thus, using a finer mesh or adaptive mesh refinement can reduce this effect. Using the adaptive

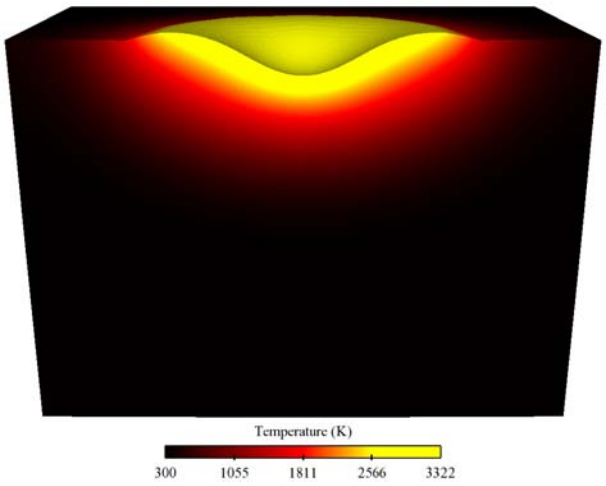

Figure 4.6. Weld pool shape and temperature profile for a 3D ALE simulation.

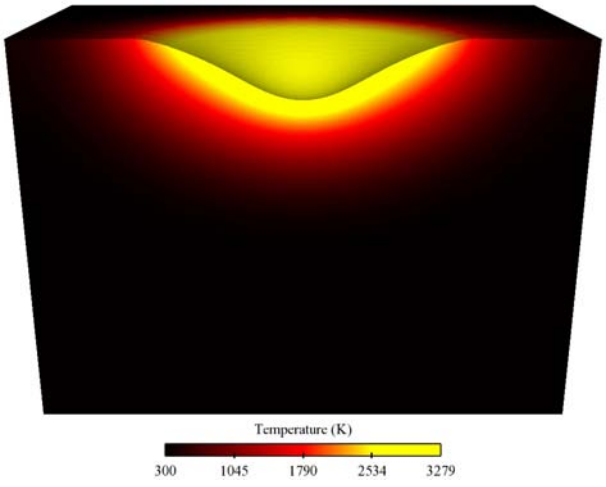

Figure 4.7. Weld pool shape and temperature profile for a 3D ALE simulation with the ambient air phase simulated.

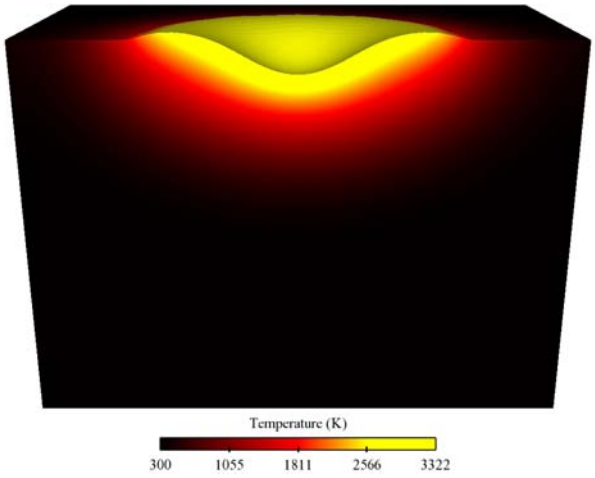

Figure 4.8. Weld pool shape and temperature profile for a 3D ALE simulation with the ambient air phase simulated. Here, the pressure is allowed to be discontinuous at the liquid/gas interface.

mesh refinement capabilities in Aria, the mesh is adapted dynamically to achieve a specified element length scale in the interface region. In figures 4.9–4.11 we keep the number of elements across the interfacial region n_α (see section 3.6) constant but reduce the interfacial thickness α. The result is that as α decreases so does the element length scale at the interface, $h_I \equiv \alpha/n_\alpha$.

Figure 4.12 quantifies the reduced mass loss of the liquid phase for both ALE and level set (LS) simulations. Here, the course mesh ALE simulation with continuous pressure demonstrates the most sever loss of mass where as the ALE simulation which accommodates the pressure discontinuity provides a near exact conservation of mass. The level set simulations demonstrate that as the element length scale at the free surface approaches zero, the mass conservation improves. Advancing the level set algorithm to accommodate a discontinuous pressure is part of the ongoing work in Aria.

4.4 Demonstration Calculation: 3D Line Weld

In this section we simply report a demonstration calculation of a traveling line weld. In this case, the laser heat flux boundary condition in equation 2.7 was extended to allow a prescribed motion of the laser source. Figure 4.13 shows the free surface position and temperature distribution for such a weld. Likewise, figure 4.14 shows

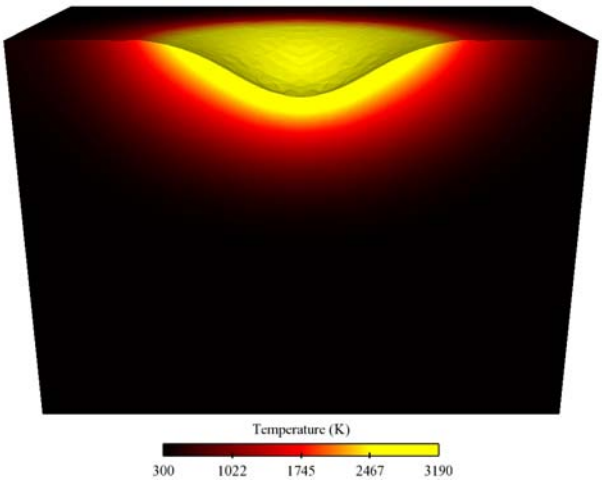

Figure 4.9. Weld pool shape and temperature profile for a 3D level set simulation with $\alpha = 4 \times 10^5$, $n_\alpha = 4$ and $h_I = 1 \times 10^{-5}$.

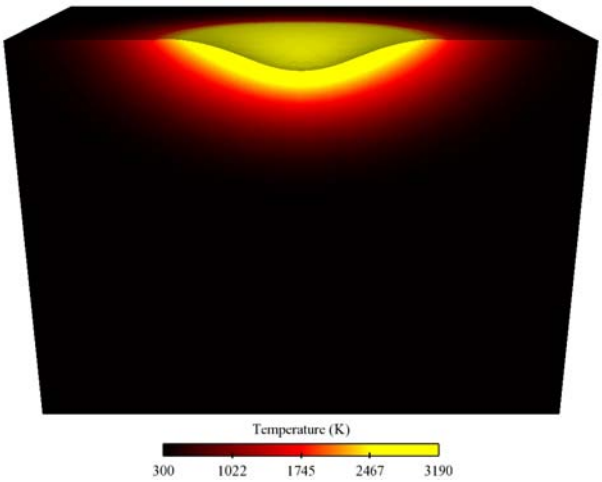

Figure 4.10. Weld pool shape and temperature profile for a 3D level set simulation with $\alpha = 2 \times 10^5$, $n_\alpha = 4$ and $h_I = 0.5 \times 10^{-5}$.

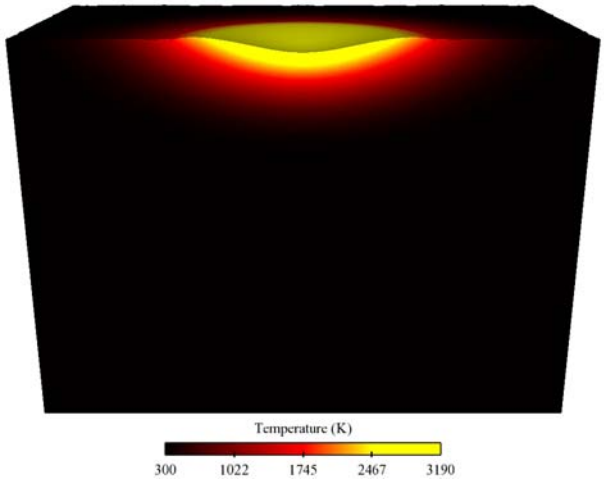

Figure 4.11. Weld pool shape and temperature profile for a 3D level set simulation with $\alpha = 1 \times 10^5$, $n_\alpha = 4$ and $h_I = 0.25 \times 10^{-5}$.

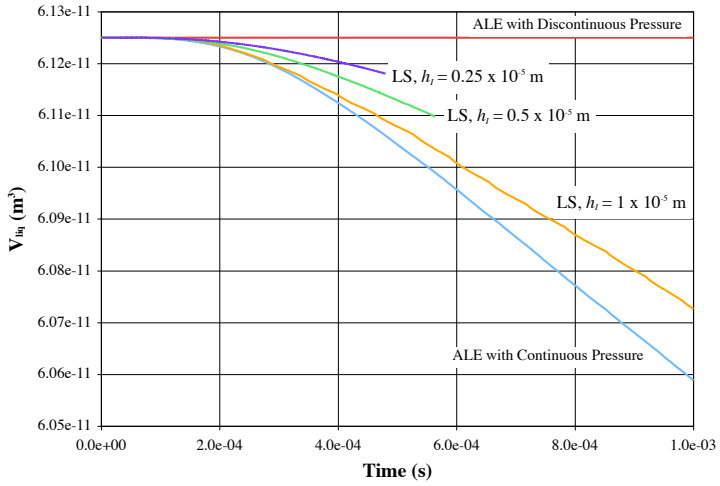

Figure 4.12. Volume of the liquid (metal) phase as a function of time as predicted by the ALE and level set (LS) simulations. Physically, the volume of this phase should be constant for all times. These results illustrate the fictitious mass loss over time due to a continuous pressure for both ALE and LS simulation with varying interfacial elements sizes.

the corresponding ALE simulation which shows excellent agreement with the level set results.

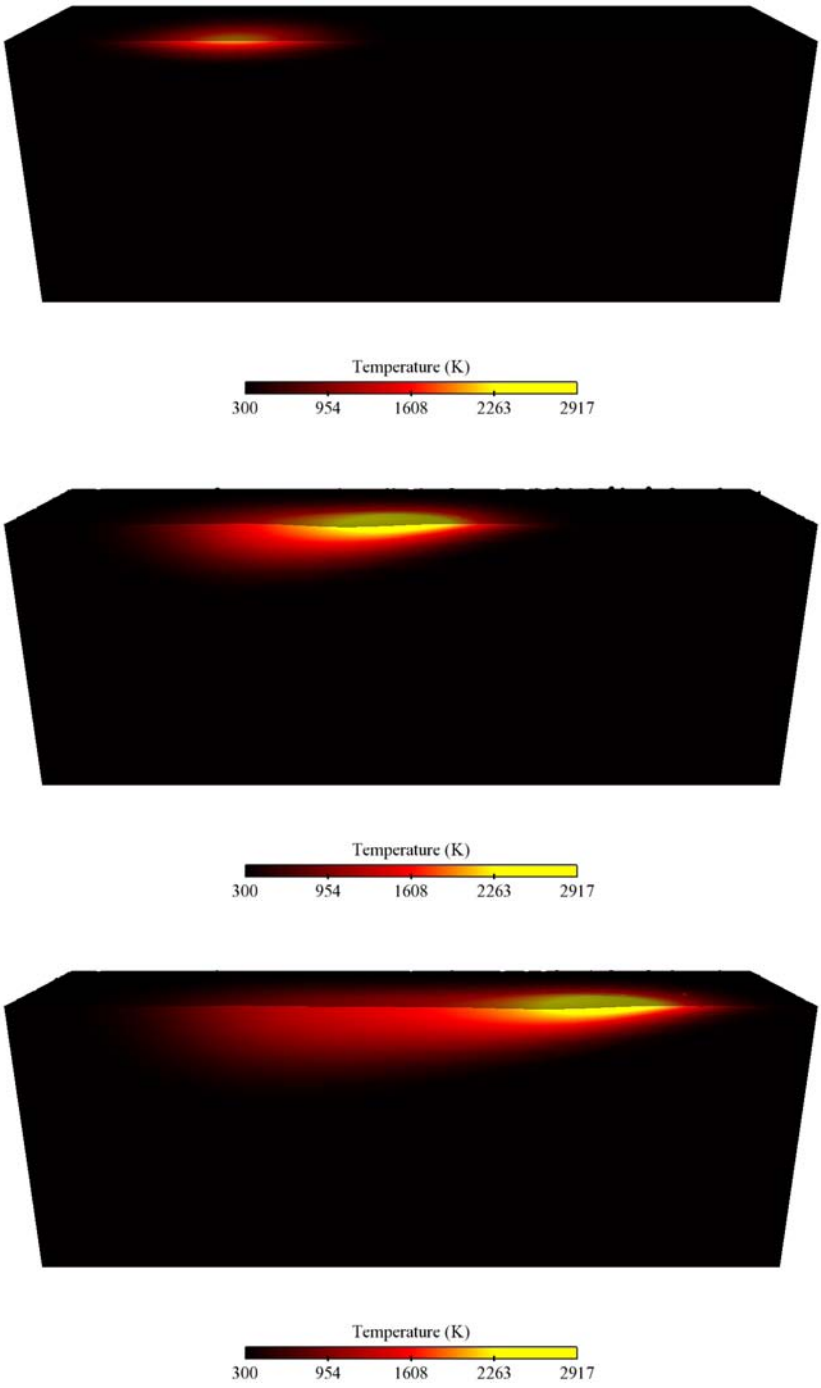

Figure 4.13. Temperature distributions and weld pool shapes at three instances during a moving line weld using the level set algorithm.

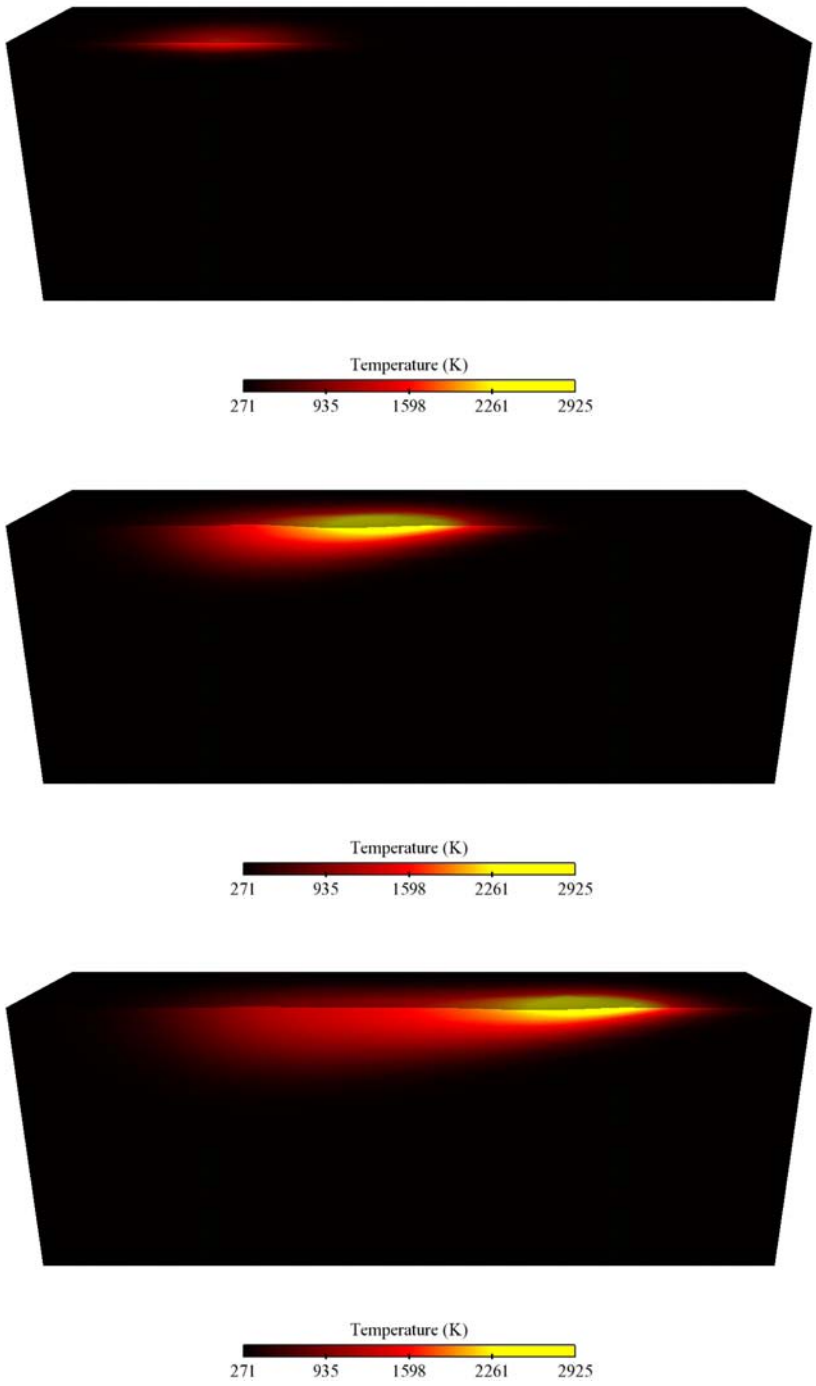

Figure 4.14. Temperature distributions and weld pool shapes at three instances during a moving line weld using the ALE algorithm.

Chapter 5

Recommendations for Future Work

In this chapter we lay forth recommendations and lessons learned during the completion of this Milestone. We make suggestions to analysts who intend to do studies similar to the work undertaken here. We also outline some future development activities that could lead to a more accurate, robust and efficient tool for problems such as laser welding.

5.1 Linear Tetrahedral Elements are Preferred

The foundational algorithms within the level set capability rely on the accurate representation and manipulation of geometric primitives such as quadrilaterals, triangles, hexahedron and tetrahedron. Two critical requirements are that these algorithms (a) remain true to the original problem discretization and (b) are convergent with mesh refinement. In cases where these two requirements are at odds with one another we choose the first. However, these two requirements can be achieved when the original mesh discretization is restricted to linear tetrahedral elements. Higher order tetrahedral basis functions can be used if the element geometries are restricted to linear (planer surfaces).

5.2 Adaptivity is Essential for Efficiency and Accuracy

As discussed in section 3.6 adaptivity is a key capability for the successful deployment of level set algorithms for large scale, 3D, free-surface flows with significant surface tension effects. Adaptivity, when combined with linear tetrahedral elements, enables the most efficient and accurate simulations for the current generation of level set algorithms available in Aria. Future algorithms may alleviate the need for adaptivity but that appears unlikely.

5.3 Support for Discontinuous Fields at Interfaces

In section 4.3.1 we described the inaccuracies that arise when the pressure is constrained to be continuous across the free surface. The ability to capture the correct pressure jump at the interface would lead to more accurate results. Moreover, the numerical artifacts often lead to spurious currents in the flow field. These numerical artifacts can cause the time integration algorithms to operate with a much smaller time step size than would otherwise be necessary and, thereby, increase the overall computational cost of a simulation.

5.4 Improved Level Set Performance

The current implementation of the level set capability in Aria could benefit from performance enhancements. Despite the efficiencies gained from utilizing the adaptive mesh refinement capability, accurate simulations still require a large number of degrees of freedom compared to the ALE algorithm. Adding support for discontinuous pressures at a free surface should help alleviate the problem. Sharp integration of level set surfaces could also reduce the number of degrees of freedom required for accurate solutions. Additional gains could be made from removing the gas phase entirely from the simulation for problems, such as laser welding, where the details there are of no interest. Lastly, the entire algorithm should be profiled for inefficiencies; course grained timing diagnostics currently available suggest there may be inefficiencies at the interface with the linear solver.

Appendix A

Example input Files

This appendix contains representative input files for performing the level set simulations of a stationary, 3D spot weld with Aria. Section A.1 contains the Cubit input ("journal") file for generating the input mesh to be used by Aria. Section A.2 contains the Aria input file for the simulation. Full details about the parameters and model details for the Aria input file can be found in the Aria User Manual (Notz et al. 2007).

A.1 Cubit Journal File

```
# SCALE = {SCALE = 0.001}
# LX    = {LX = 0.7}
# LY    = {LY = 0.7}
# LZ_BOT = {LZ_BOT = 0.5}
# LZ_TOP = {LZ_TOP = 0.1}

reset
set associativity complete on
create brick width {LX/2} depth {LY/2} height {LZ_BOT}
body 1 move {LX/4} {LY/4} {-LZ_BOT/2}

create brick width {LX/2} depth {LY/2} height {LZ_TOP}
body 2 move {LX/4} {LY/4} {LZ_TOP/2}
merge all

#curve all interval 15
#curve 3 interval 30
#curve 4 interval 30
#curve 9 interval 30

surface 1 size {0.01*2*0.7}
surfa all scheme trimesh
mesh surface 1
```

```
surface  3 sizing function type bias start curve 4 factor 1.1
surface  9 sizing function type bias start curve 4 factor 1.1
surface  4 sizing function type bias start curve 3 factor 1.1
surface 10 sizing function type bias start curve 3 factor 1.1
surface  5 sizing function type bias start curve 2 factor 1.1
surface 11 sizing function type bias start curve 2 factor 1.1
surface  6 sizing function type bias start curve 1 factor 1.1
surface 12 sizing function type bias start curve 1 factor 1.1

mesh surface 3 4 5 6 9 10 11 12

sideset 1 surface 4 10 # x = xmin
sideset 2 surface 6 12 # x = xmax
sideset 3 surface 3  9 # y = ymin
sideset 4 surface 5 11 # y = ymax
sideset 5 surface 2    # z = zmin
sideset 6 surface 7    # z = zmax

volume all scheme tetmesh
mesh volume all
merge all

block 1 volume 1 2

transform mesh output scale {SCALE}

export genesis "spot3d_quarter_graded_mesh_T1.exoII" overwrite
```

A.2 Aria Input File

```
#
# alpha          = {alpha=2.e-5}
# beam_radius    = {beam_radius = 0.0001}
# elem_per_alpha = {elem_per_alpha = 4}
#
Begin Sierra 3D_ALE_Laser_Spot-Welding_Test

   Begin Aria Material multiphase
      Level Set Heaviside = Smooth
      Level Set Width     = Constant width={alpha}
      Surface Tension     = Linear_T sigma0= 1.943 dsigmadT=-4.3e-4 T_ref=1809
      thermal conductivity = Phase_Average
      Viscosity            = Phase_Average
```

```
End

Begin Aria Material Stainless # Units in mks
   Density              = Constant Rho    = 7900.
   Viscosity            = Weld c0=0.15616 c1=-3.3696e-5 c2=1.0191e-8 \$
      c3=-1.0413e-12 T_liq=1623 T_90=1528 T_max=4000
   Specific Heat        = Polynomial Variable=Temperature Order=1 C0=425.75 \$
      C1=170.833e-3
   Thermal Conductivity = Thermal A =10.7143  B=14.2857e-3 C=0 D=0
   Momentum Stress      = Incompressible_Newtonian
   Momentum Stress      = LS_Capillary
   Heat Conduction      = Fouriers_Law
End

Begin Aria Material air
   Density              = Constant Rho = 1
   Viscosity            = Constant mu = {2.e-5 * 1000}
   Thermal Conductivity = Constant K = 0.025
   Specific Heat        = Constant Cp = 1000
   Momentum Stress      = Incompressible_Newtonian
   Momentum Stress      = LS_Capillary
   Heat Conduction      = Fouriers_Law
End

Begin Aztec Equation Solver gmres_ilut
   solution method        = gmres
   preconditioning method = dd-ilut
   maximum iterations     = 500
   param-int AZ_kspace value 500
   residual norm tolerance = 1.e-6
   param-real AZ_drop value 1e-8
   param-int AZ_ilut_fill value 2
   preconditioner subdomain overlap = 2
   matrix reduction = fei-remove-slaves
End

Begin Finite Element Model The_Model
   Database Name = spot3d_quarter_graded_mesh_T1.exoII
   Begin Parameters For Block block_1
      material multiphase
      phase a = stainless
      phase b = air
   End
End
```

```
Begin Adaptivity Controller my_adaptive_strategy
   Max Outer Adapt Steps is 1
   Max Inner Adapt Steps is 1
End

Begin Zoltan Parameters Zoltan_Params
   Load Balancing Method = Hilbert Space Filling Curve
   Over Allocate Memory  = 1.5
   Reuse Cuts            = true
   Algorithm Debug Level = 2
   Check Geometry        = true
   Zoltan Debug Level    = 2
   Timer                 = cpu
   Debug Memory          = 0
End

Begin Procedure The_Procedure

   Begin Solution Control Description
      Use System Main
      Begin System Main
         Simulation Start Time           = 0.000
         Simulation Termination Time     = 0.001
         Simulation Max Global Iterations = 2000
         Begin Transient The_Time_Block
            Advance The_Region
            Event LS_CONSTRAINED_REDISTANCE when "(CURRENT_STEP - \$
               LAST_LS_CONSTRAINED_REDISTANCE_STEP) >= 25 || \$
               (LS_GRADIENT_ERROR_NORM(0.) > 0.1 && (CURRENT_STEP - \$
               LAST_LS_CONSTRAINED_REDISTANCE_STEP) >= 5)"
            Event LS_COMPUTE_SIZES
         End
      End

      Begin Parameters For Transient The_Time_Block
         Start Time = 0.0
         Begin Parameters For Aria Region The_Region
            Initial Time Step Size          = 1.e-6
            Maximum Time Step Size          = 5.e-6
            Minimum Time Step Size          = 1.e-9
            Minimum Resolved Time Step Size = 1.e-8
            Time Step Variation             = Adaptive
            Predictor-Corrector Tolerance   = 0.01
            Courant Limit                   = 0.5
         End
```

```
   End
End

Begin Aria Region The_Region

   Use Finite Element Model The_Model

   Use Adaptivity Controller my_adaptive_strategy
   Adapt Mesh on Isovolume Level_Set h={h=alpha/elem_per_alpha} \$
      min={-(alpha/2+1.5*h)} max={alpha/2+1.5*h}
   Enable Rebalance with Threshold = 1.2 Using Zoltan Parameters \$
      Zoltan_Params
   Rebalance Load Measure Type = Constant

   Use Linear Solver gmres_ilut

   Predictor Fields = Velocity
   Predictor Fields = Temperature
   Predictor Fields = Not Pressure

   Nonlinear Solution Strategy  = Newton
   Maximum Nonlinear Iterations = 10
   Nonlinear Residual Tolerance = 1.0e-6
   Nonlinear Relaxation Factor  = 1.0
   nonlinear correction tolerance = 0

   EQ Level_Set for Level_Set on block_1 Using Q1 with Mass Adv
   IC Linear on block_1 Level_Set COEFF = 0.0 0.0 0.0 1.0
   Advection Velocity for Level_Set = OLD_VELOCITY

   Begin Level Set Interface LS
      Distance Variable = solution->LEVEL_SET
      Velocity Variable = solution->VELOCITY
      Narrow Band Width = {4.0*alpha}
   EnD

   EQ Momentum_A For Velocity On block_1 Using Q1 With Mass Adv Diff Src
   EQ Momentum_B For Velocity On block_1 Using Q1 With Mass Adv Diff Src
   Pressure Stabilization Is PSPP_CONSTANT With Scaling = 1.0

   EQ Continuity_A For Pressure On block_1 Using Q1 With Div
   EQ Continuity_B For Pressure On block_1 Using Q1 With Div

   BC Const Dirichlet At surface_1 Velocity_X = 0.0
```

```
BC Const Dirichlet At surface_2 Velocity_X = 0.0
BC Const Dirichlet At surface_2 Velocity_Y = 0.0
BC Const Dirichlet At surface_2 Velocity_Z = 0.0

BC Const Dirichlet At surface_3 Velocity_Y = 0.0

BC Const Dirichlet At surface_4 Velocity_X = 0.0
BC Const Dirichlet At surface_4 Velocity_Y = 0.0
BC Const Dirichlet At surface_4 Velocity_Z = 0.0

BC Const Dirichlet At surface_5 Velocity_X = 0.0
BC Const Dirichlet At surface_5 Velocity_Y = 0.0
BC Const Dirichlet At surface_5 Velocity_Z = 0.0

Source For Momentum_A On block_1 = Hydrostatic gx=0.0 gy=0.0 gz=-9.81
Source For Momentum_B On block_1 = Hydrostatic gx=0.0 gy=0.0 gz=-9.81

EQ Energy_A for Temperature on block_1 using Q1 with Mass Adv Diff
EQ Energy_B for Temperature on block_1 using Q1 with Mass Adv Diff

IC Const on block_1 Temperature = 300.
BC Const Dirichlet on surface_5 Temperature = 300.

BC Diffuse for Momentum_A on surface_AB = Vapor_Recoil_Pressure \$
   TBOIL = 3000.
BC Diffuse for Momentum_B on surface_AB = Vapor_Recoil_Pressure \$
   TBOIL = 3000.
BC Diffuse for Energy_A on surface_AB = Gaussian_Spot_Weld src_x=0 \$
   src_y=0 src_z=1 dir_x=0 dir_y=0 dir_z=-1 R={beam_radius} \$
   Flux = 2.546e9 R_EFF = 0.6
BC Diffuse for Energy_B on surface_AB = Gaussian_Spot_Weld src_x=0 \$
   src_y=0 src_z=1 dir_x=0 dir_y=0 dir_z=-1 R={beam_radius} \$
   Flux = 2.546e9 R_EFF = 0.6
BC Diffuse for Energy_A on surface_AB = Rad TO=300 CRAD=5.67e-8
BC Diffuse for Energy_B on surface_AB = Rad TO=300 CRAD=5.67e-8
BC Diffuse for Energy_A on surface_AB = Vapor_Cooling Tboil=3000.
BC Diffuse for Energy_B on surface_AB = Vapor_Cooling Tboil=3000.

Postprocess Viscosity  on block_1
Postprocess thermal_conductivity  on block_1

Begin Results Output The_Output
   Database Name = soln.e
   At Step 0, Increment is 10
   Title Spot Weld
```

```
            Nodal Variables = solution->Level_Set          as F
            Nodal Variables = solution->Velocity           as V
            Nodal Variables = solution->Pressure           as P
            Nodal Variables = solution->Temperature        as T
            Nodal Variables = pp->Viscosity                as MU
            Nodal Variables = pp->thermal_conductivity     as K
         End
      End
   End
End
```

References

Ted Belytschko, Wing Kam Liu, and Brian Moran. *Nonlinear Finite Elements for Continua and Structures*. John Wiley and Sons, 2004. 3.3

Javier Bonet and Richard D. Wood. *Nonlinear Continuum Mechanics for Finite Element Analysis*. Cambridge University Press, 1997. 3.3

R. A. Cairncross, P. R. Schunk, T. A. Baer, R. R. Rao, and P. A. Sackinger. A finite element method for free surface flows of incompressible fluids in three dimensions. part i. boundary fitted mesh motion. *Int. J. Numer. Methd. Fluids*, 33:375–403, 2000. 2.3.5, 3.1, 3.3

Clark R. Dohrmann and Pavel B. Bochev. A stabilized finite element method for the Stokes problem based on polynomial pressure projections. *Int. J. Num. Meth. Fluids*, 46:183–201, 2004. 3.5

H. C. Edwards. SIERRA Framework version 3: core theory and design. Technical Report SAND2002-3616, Sandia National Laboratories, 2002. 3.6

C. W. Hirt and B. D. Nicholls. Volume of fluid (VOF) method for dynamics of free boundaries. *J. Comp. Phys.*, 39:201–225, 1981. 3.1

T. J. R. Hughes. *The Finite Element Method*. Dover Publications, New York, USA, 2000. 3.4, 3.5

M. P. Kanouff. Vapor recoil pressure II. Technical Report Internal Memorandum, Sandia National Laboratories, Albuquerque, NM, USA, March 2003. 2.3.3, 2.3.4

C. J. Knight. Theoretical modeling of rapid surface vaporization with back pressure. *AIAA Journal*, 17(5):519–523, 1979. 2.3.3, 2.3.4

P. K. Notz, S. R. Subia, M. M. Hopkins, H. K. Moffat, and D. R. Noble. Aria 1.6 user manual. In preparation, 2007. A

J. A. Sethian. *Level Set Methods and Fast Marching Methods*, volume 3 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, New York, USA, 2nd edition, 1999. 3.1

J. R. Steward and H. C. Edwards. SIERRA Framework version 3: h-adaptivity design and use. Technical Report SAND2002-4016, Sandia National Laboratories, 2002. 3.6

M. Tanaka and J. J. Lowke. Predictions of weld pool profiles using plasma physics. *J. Phys. D: Appl. Phys.*, 40:R1–R23, 2007. 4.2

DISTRIBUTION:

1 MS 0139
A. L. Hale, 1900

1 MS 0380
H. S. Morgan, 1540

1 MS 0382
S. E. Gianoulakis, 1541

1 MS 1411
J. S. Lash, 1514

1 MS 0836
R. A. Roach, 1814

3 MS 0836
P. K. Notz, 1514

3 MS 0836
M. J. Martinez, 1514

3 MS 0836
D. R. Noble, 1514

1 MS 0847
C. R. Dohrmann, 1523

2 MS 9018
Central Technical Files, 8944

2 MS 0899
Technical Library, 9536

Sandia National Laboratories